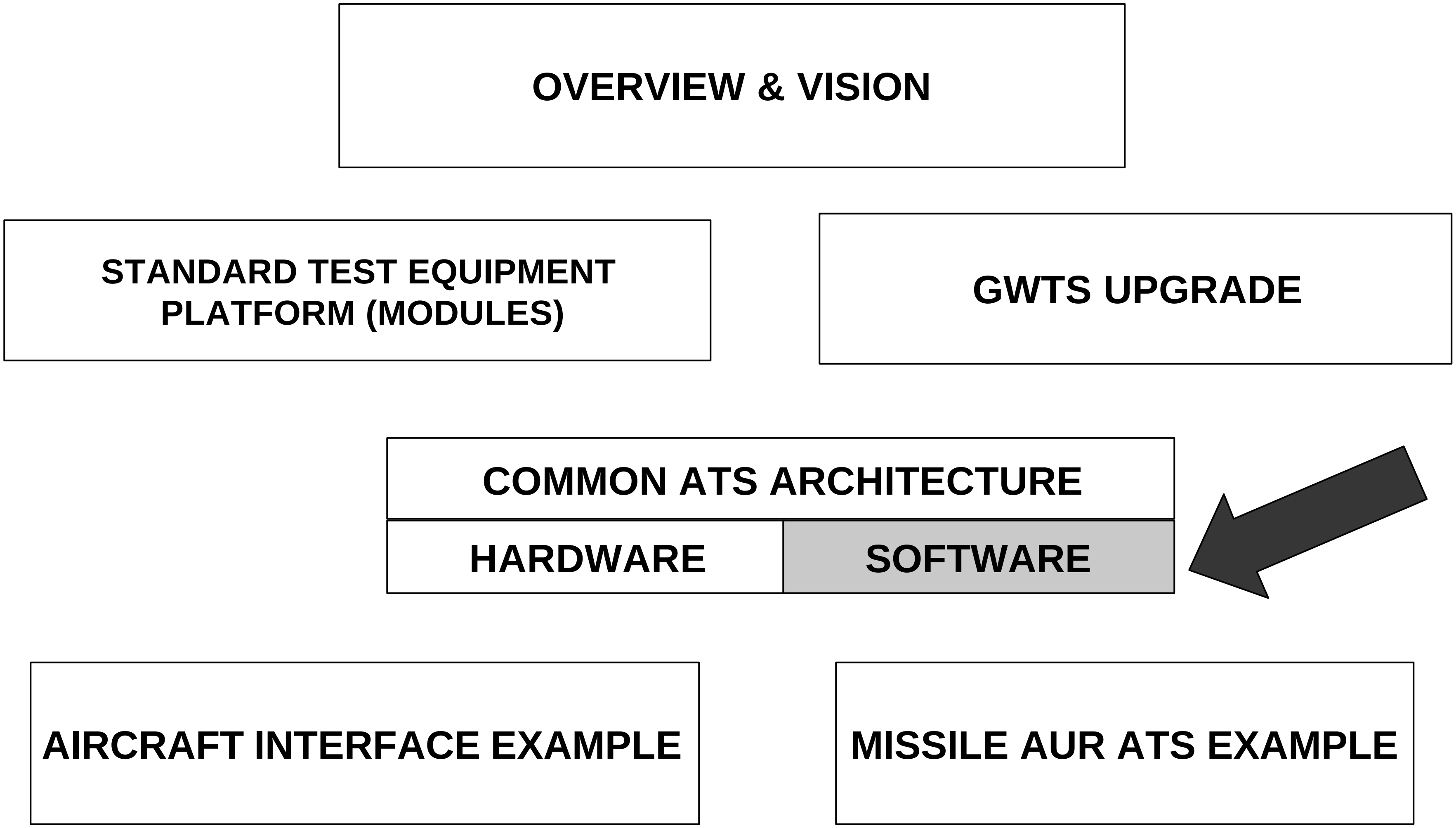
SESSION PAPERS



OVERVIEW & VISION

STANDARD TEST EQUIPMENT PLATFORM (MODULES)

GWTS UPGRADE

COMMON ATS ARCHITECTURE

HARDWARE

SOFTWARE

AIRCRAFT INTERFACE EXAMPLE

MISSILE AUR ATS EXAMPLE

A Software Approach for a Common Munitions Test Architecture

Introduction and the Software Approach
Ira Lieberman
Test Automation, Inc.

The Test Executive and Data Logging Methods
Stephen Insalaco
Raytheon Missile Systems

The Application's Software Architecture and Summary
Alfred A. Van Oosbree
Raytheon Missile Systems

Introduction and the Software Approach

Ira Lieberman

Test Automation, Inc.

Why Pursue a Standard Architecture

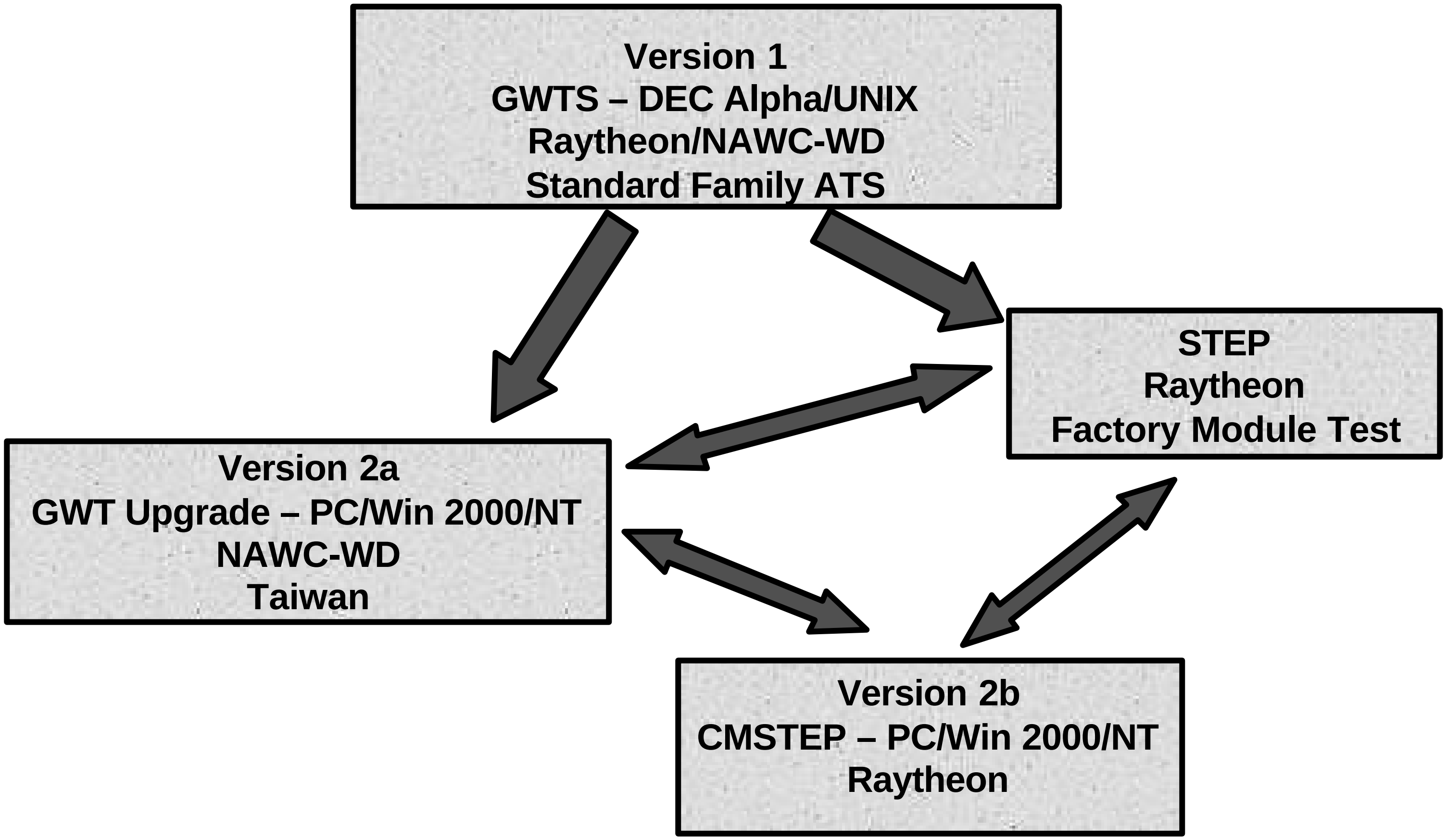
Raytheon

- Design for Obsolescence
- Modular Design Philosophy
- Module Reuse
- Common Support Framework

Motivation for a Standard Software Architecture

- A Standard Architecture facilitates changes to software to accommodate:
  - ATE hardware changes due to:
    - Instrument obsolescence
    - Flexible ATE configurations
  - UUT test requirements changes due to:
    - Incremental development
    - Product enhancements
- It also provides
  - Common operator interfaces
  - Maximize software reuse
  - Minimize training costs

Software Architecture Interaction



Version 1
GWTS – DEC Alpha/UNIX
Raytheon/NAWC-WD
Standard Family ATS

STEP
Raytheon
Factory Module Test

Version 2a
GWT Upgrade – PC/Win 2000/NT
NAWC-WD
Taiwan

Version 2b
CMSTEP – PC/Win 2000/NT
Raytheon

GWTS Software Architecture

Raytheon

- The GWTS Upgrade provides a PC/Windows transformation of the DEC/UNIX GWTS design
- Approximately 400k lines of software are being rehosted
- Some new software development utilities are being employed
- Wherever possible, source code is being rehosted rather than rewritten
- The paper provides more details
- The hardware aspects are discussed in Mr. Stanfield’s paper

The Focus of the Presentations

Raytheon

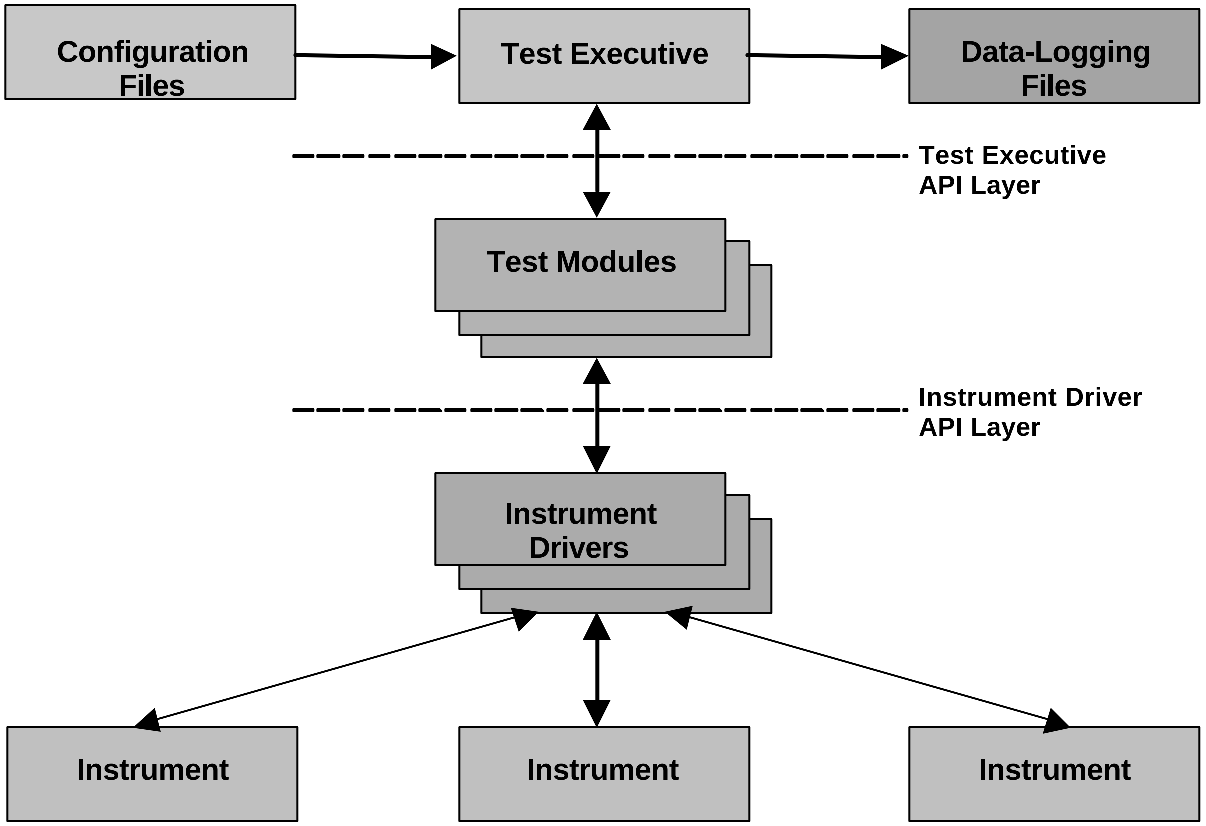
- A three-component software architecture
  - Test executive
  - Test programs
  - Instrument drivers
- A test executive that:
  - Insulation of the test executive from the test program instrument layer
  - Is easy to learn & use, promotes independent/module development
- A common test program architecture featuring:
  - High-level test oriented programming language
  - lower-level Application Program Interface (API)
  - Isolation of the run-time environment from the development environment
- Instrument drivers that are:
  - Reusable
  - Independent

The Test Executive and Data Logging Methods

Stephen Insalaco

Raytheon Missile Systems

The Test Executive Software Architecture



The Test Executive

Raytheon

- Test program selection and test sequencing services
- Test flow control services (start, stop, looping, shutdown, abort)
- Test data management services (data-logging)
- Test validation services(parameter limit files)
- Other considerations:
  - Kernel of the test software
  - Well implemented set of code modules
  - Utilizes recurring software modules

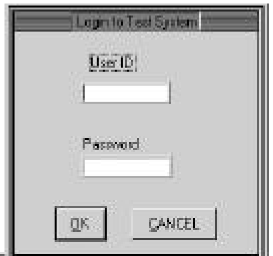
Benefits of the Test Executive Approach

- Standardizes on a common programming software architecture.
- Provides a common set of pre-programmed services and interfaces (API interface)
- Injects proven and existing code modules into new test programs (promotes code reuse)
- Reduces test program developer learning time
- Reduces test operator learning time
- Provides an organizational competitive advantage

Test Executive Services

Raytheon

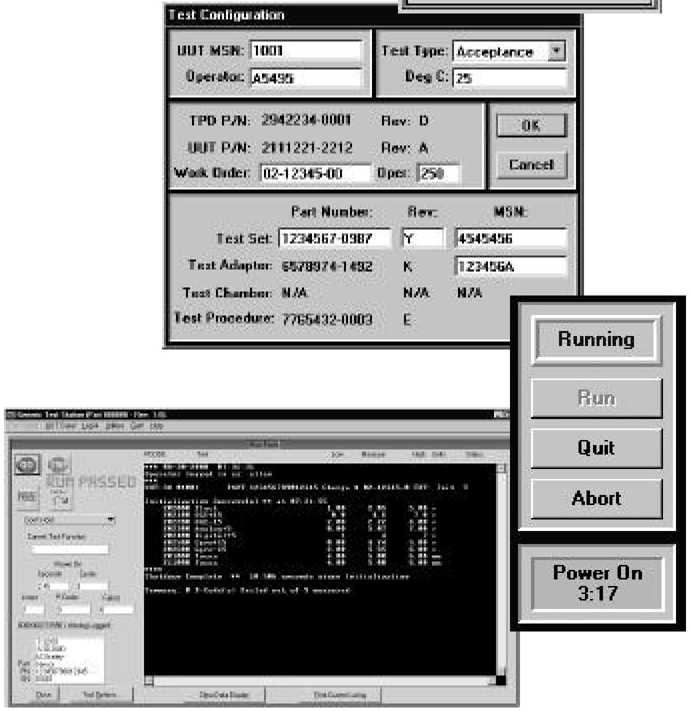
- User authentication and privilege services
- Unit under test (UUT) data tracking
- Test scheduling (presentation and launching)
- Test flow control
  - Initialization and shutdown
  - Start, stop, continue, abort
  - Looping
  - Timer
- Data management services



Login to Test System

User ID:

Password:



Test Configuration

UUT MSN: 1001

Operator: A5495

Test Type: Acceptance

Deg C: 25

TPD P/N: 2942234-0001 Rev: D

UUT P/N: 2111221-2212 Rev: A

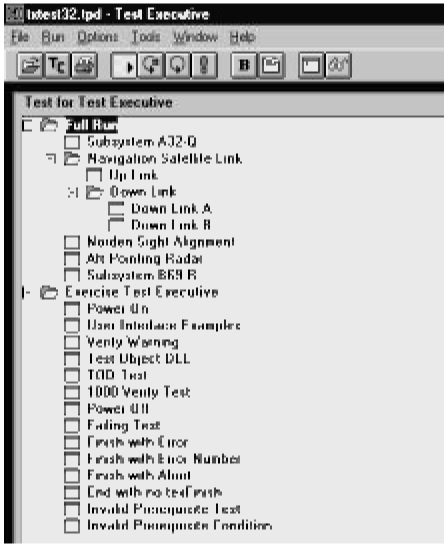
Work Order: 02-12345-00 Oper: 250

	Part Number:	Rev:	MSN:
Test Set:	1234567-0987	Y	4545456
Test Adapter:	6578974-1492	K	123456A
Test Chamber:	N/A	N/A	N/A
Test Procedure:	7765432-0003	E	

Test Executive Features

Raytheon

- **Application “*bells and whistles*” that allows for program variation**
- **Configure test program for expected behavior**
- **Features can be toggled between ENABLED & DISABLED states and associated with user privilege levels**
- **Typical features may include:**
  - **Segmented testing strategies**
  - **Go-No-Go testing strategies**
  - **Conditional branching and looping strategies**
  - **Parameter-specific forced PASS/FAIL testing**



- Full Run
  - □ Subsystem A32-Q
  - Navigation Satellite Link
    - □ Up Link
    - Down Link
      - □ Down Link A
      - □ Down Link B
  - □ Norden Sight Alignment
  - □ Aft Pointing Radar
  - □ Subsystem B69 R
- Exercise Test Executive
  - □ Power On
  - □ User Interface Examples
  - □ Verify Warning
  - □ Test Object DLL
  - □ TOD Test
  - □ 1000 Verify Test
  - □ Power Off
  - □ Failing Test
  - □ Finish with Error
  - □ Finish with Error Number
  - □ Finish with Abort
  - □ End with no testFinish
  - □ Invalid Prerequisite Test
  - □ Invalid Prerequisite Condition

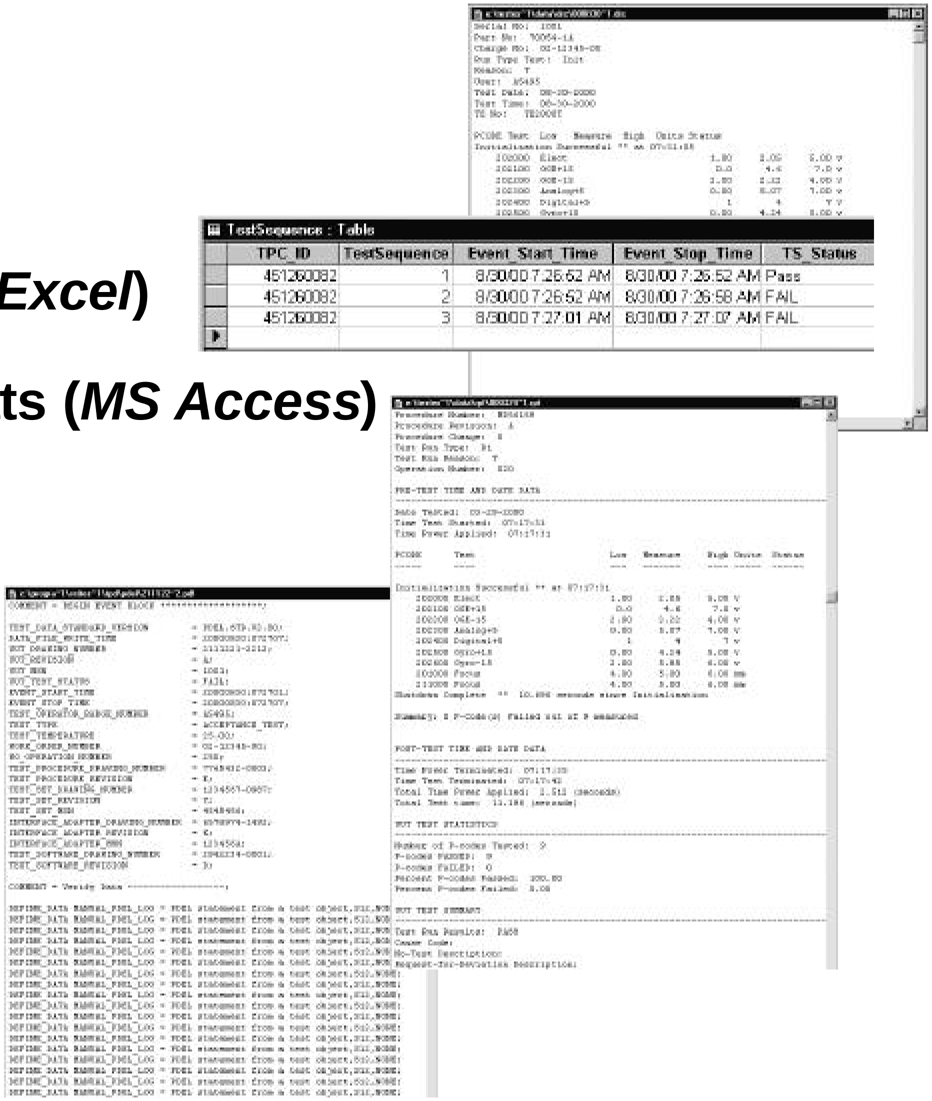
Data Logging

- Provides a permanent record of test activities
- Electronic and/or hardcopy container for test results
- Typically part of the UUT assembly record history
- Feeds organizational query and report generation tools
- Forms the basis for further analysis
- Optimize on one output format

Data Logging Methods

Raytheon

- **Data record sheets**
- **Spreadsheet formats (*MS Excel*)**
- **Relational database formats (*MS Access*)**
- **IEEE-1545 ¬**
- **Proprietary formats**



```
Serial No:  1081
Part No:  70054-16
Charge No:  02-12345-08
Run Type Test:  Init
Reason:  T
User:  45485
Test Date:  08-30-2000
Test Time:  08-30-2000
TS No:   TE2000T

PCODE Test  Low   Measure  High  Units Status
Initialization Successful ** at 07:11:18
     102000  Elect                         1.80      2.05      5.00 v
     102100  OCE+15                         0.0       4.6       7.0 v
     102200  OCE-15                        1.80      2.22      4.00 v
     102300  Analog+5                      0.80      5.07      7.00 v
     102400  Digital+5                        1         4         7 v
     102500  Gyro+15                       0.80      4.24      5.00 v
```

TestSequence : Table

TPC_ID	TestSequence	Event_Start_Time	Event_Stop_Time	TS_Status
451260082	1	8/30/00 7:26:52 AM	8/30/00 7:26:52 AM	Pass
451260082	2	8/30/00 7:26:52 AM	8/30/00 7:26:58 AM	FAIL
451260082	3	8/30/00 7:27:01 AM	8/30/00 7:27:07 AM	FAIL

```
Procedure Number:  8254148
Procedure Revision:  A
Procedure Change:  8
Test Run Type:  R1
Test Run Reason:  T
Operation Number:  020

PRE-TEST TIME AND DATE DATA
--------------------------------------------------------------------------
Date Tested:  08-29-2000
Time Test Started:  07:17:31
Time Power Applied:  07:17:31

PCODE      Test                          Low   Measure    High Units  Status
-----      ----                          ---   -------    ---- -----  ------

Initialization Successful ** at 07:17:31
   102008 Elect                         1.80     2.88     5.08 V
   102108 OCE+15                         0.0      4.6      7.8 v
   102208 OCE-15                        2.80     3.22     4.08 V
   102308 Analog+5                      0.80     5.07     7.08 V
   102408 Digital+5                        1        4        7 v
   102508 Gyro+15                       0.80     4.24     5.08 V
   102608 Gyro-15                       2.80     3.88     4.08 v
   103008 Focus                         4.80     5.80     6.08 mm
   113008 Focus                         4.80     5.80     6.08 mm
Shutdown Complete  **  10.888 seconds since Initialization

Summary: 0 P-Code(s) Failed out of 9 measured


POST-TEST TIME AND DATE DATA
--------------------------------------------------------------------------
Time Power Terminated:  07:17:33
Time Test Terminated:  07:17:42
Total Time Power Applied:  1.512 (seconds)
Total Test time:  11.188 (seconds)

UUT TEST STATISTICS
--------------------------------------------------------------------------
Number of P-codes Tested:  9
P-codes PASSED:  9
P-codes FAILED:  0
Percent P-codes Passed:  100.00
Percent P-codes Failed:  0.00

UUT TEST SUMMARY
--------------------------------------------------------------------------
Test Run Results:  PASS
Cause Code:
No-Test Description:
Request-for-Deviation Description:
```

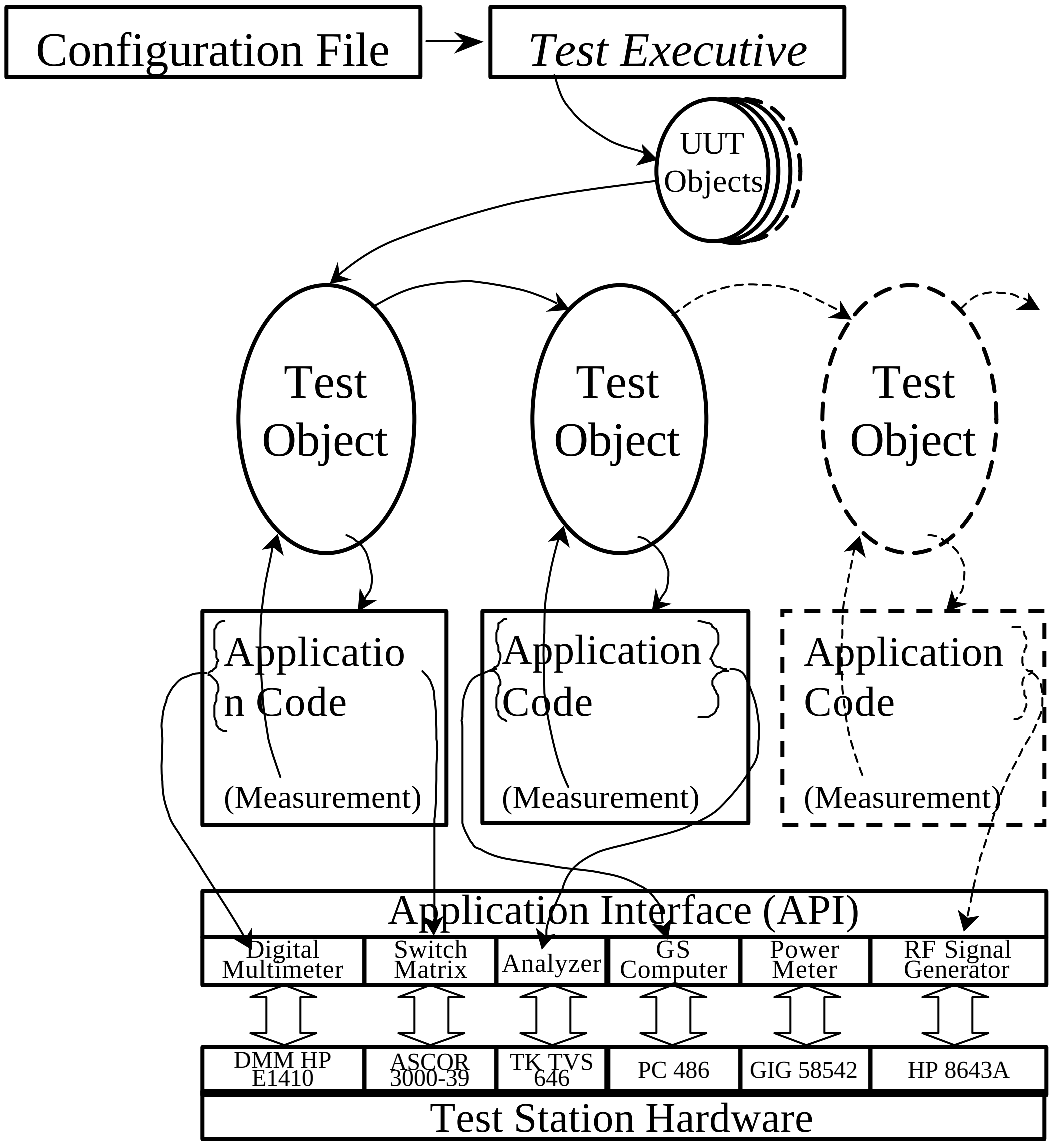
```
COMMENT = BEGIN EVENT BLOCK ********************;

TEST_DATA_STANDARD_VERSION          = PDEL,STD,92,00;
DATA_FILE_WRITE_TIME                = 20000830:072707;
UUT_DRAWING_NUMBER                  = 1111111-2212;
UUT_REVISION                        = A;
UUT_SER                             = 1081;
UUT_TEST_STATUS                     = FAIL;
EVENT_START_TIME                    = 20000830:072701;
EVENT_STOP_TIME                     = 20000830:072707;
TEST_OPERATOR_BADGE_NUMBER          = 45485;
TEST_TYPE                           = ACCEPTANCE_TEST;
TEST_TEMPERATURE                    = 25.00;
WORK_ORDER_NUMBER                   = 02-12345-80;
WO_OPERATION_NUMBER                 = 250;
TEST_PROCEDURE_DRAWING_NUMBER       = 7745412-0803;
TEST_PROCEDURE_REVISION             = K;
TEST_SET_DRAWING_NUMBER             = 1234567-0807;
TEST_SET_REVISION                   = T;
TEST_SET_SER                        = 4848484;
INTERFACE_ADAPTER_DRAWING_NUMBER    = 8978974-1482;
INTERFACE_ADAPTER_REVISION          = K;
INTERFACE_ADAPTER_SER               = 1234568;
TEST_SOFTWARE_DRAWING_NUMBER        = 1048234-0801;
TEST_SOFTWARE_REVISION              = B;

COMMENT = Verify Data --------------------;

DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
DEFINE_DATA MANUAL_PDEL_LOG = PDEL statement from a test object,512,NONE;
```

The Application's Software Architecture and Summary

Alfred A. Van Oosbree

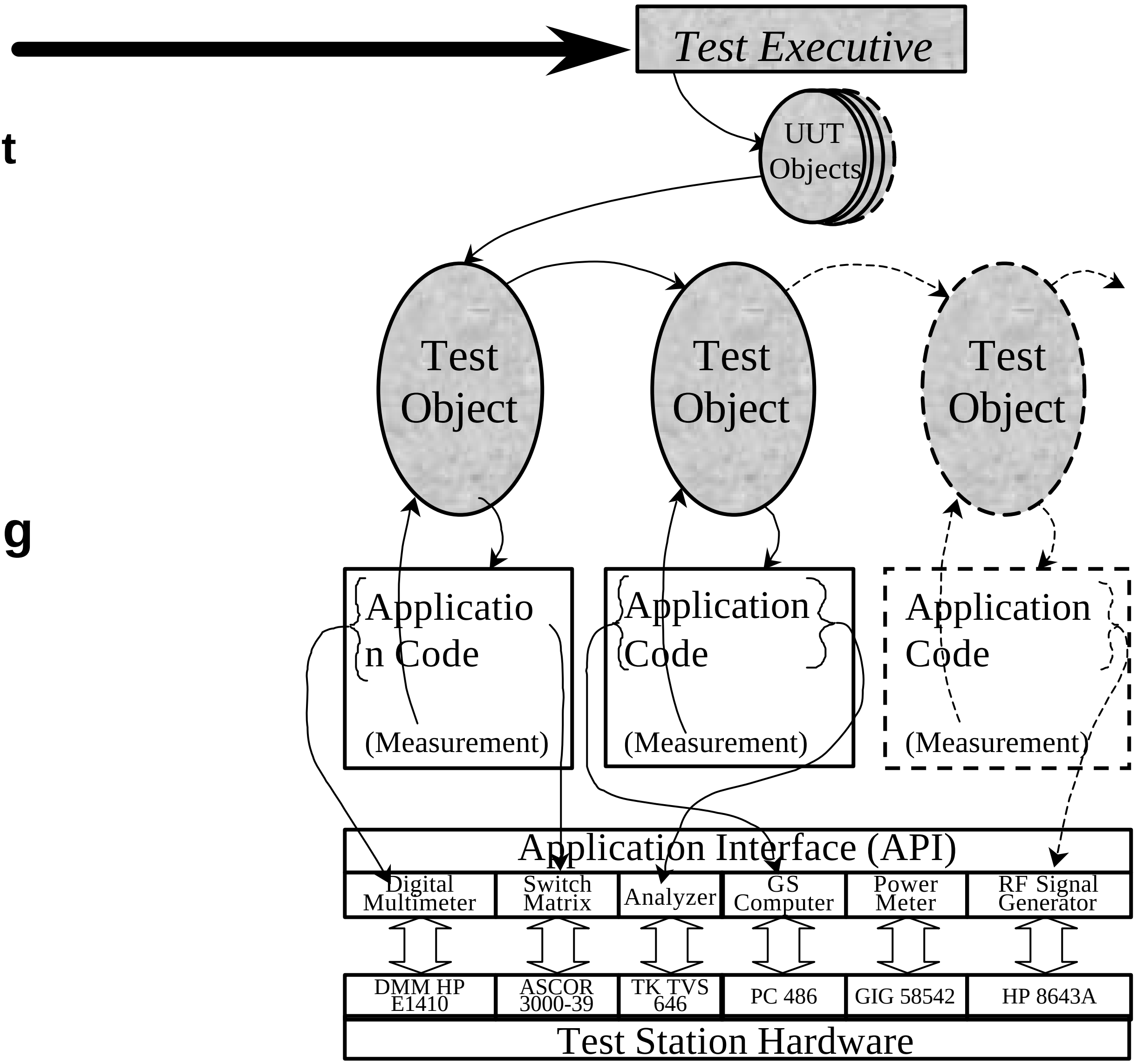
Raytheon Missile Systems

Architecture Overview



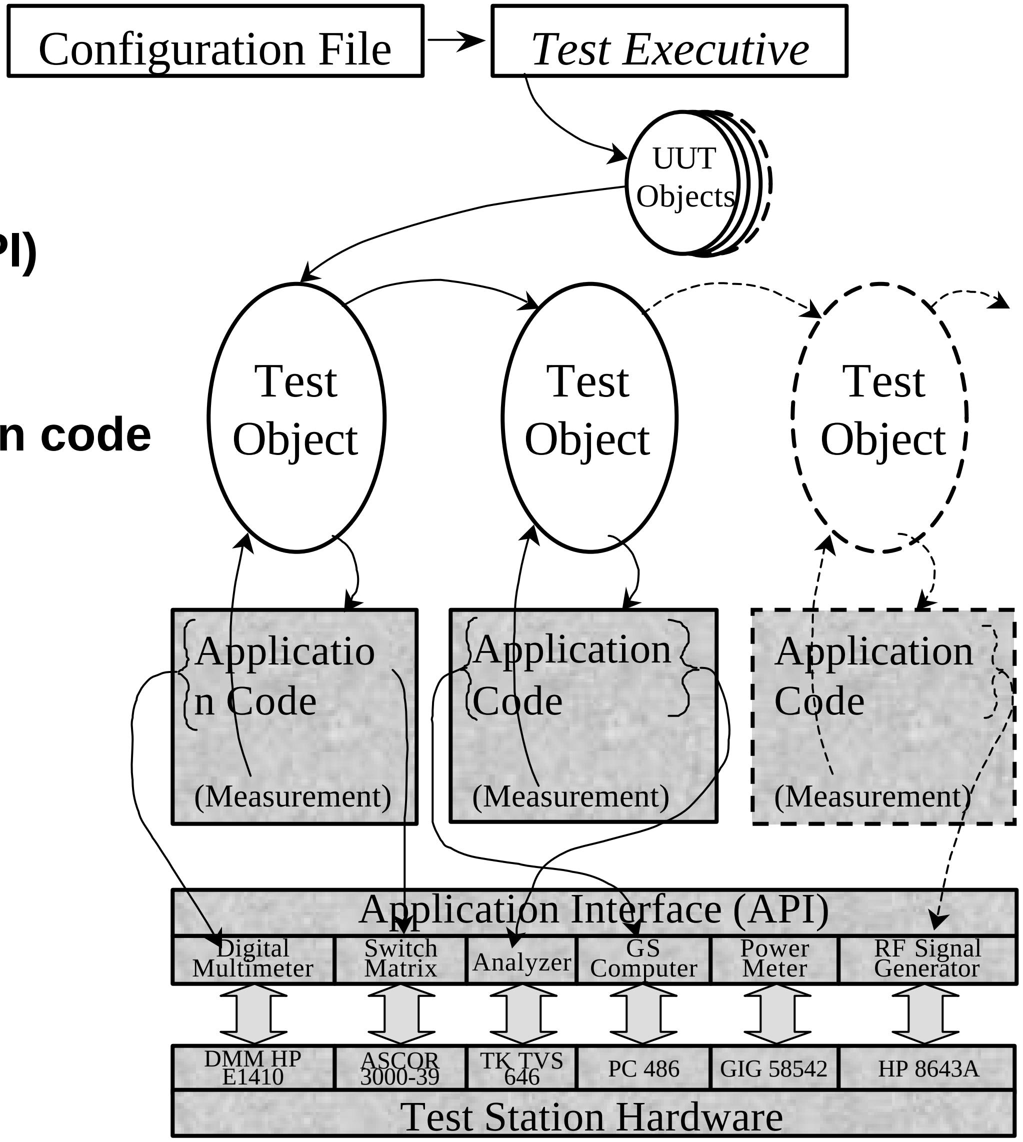
The Executive/Application Link

- **Configuration File**
  - **Describes each test**
  - **Excel file**
  - **Access file**
  - **Text based**
- **Executive processing instantiates:**
  - **UUT Object(s)**
  - **Test Objects**



Test Code Structure

- **Application Code**
  - **Code isolated from H/W (API)**
    - **Obsolescence**
  - **Complete set of initialization code**
  - **Object Oriented**



Test Structure

- **Advantages**
  - **Conditions readily visible to the programmer.**
  - **Development and maintenance are enhanced.**
  - **No guessing about the current state of the test.**
  - **Regardless of placement, code properly executes.**
  - **If test requirements change, it’s easy to pinpoint the exact code used to control the affected tests.**
    - **Concurrent Engineering - no whining allowed!**
    - **Diagnostics**
- **Disadvantages**
  - **Lot of repeated calls**
  - **Slows the test**

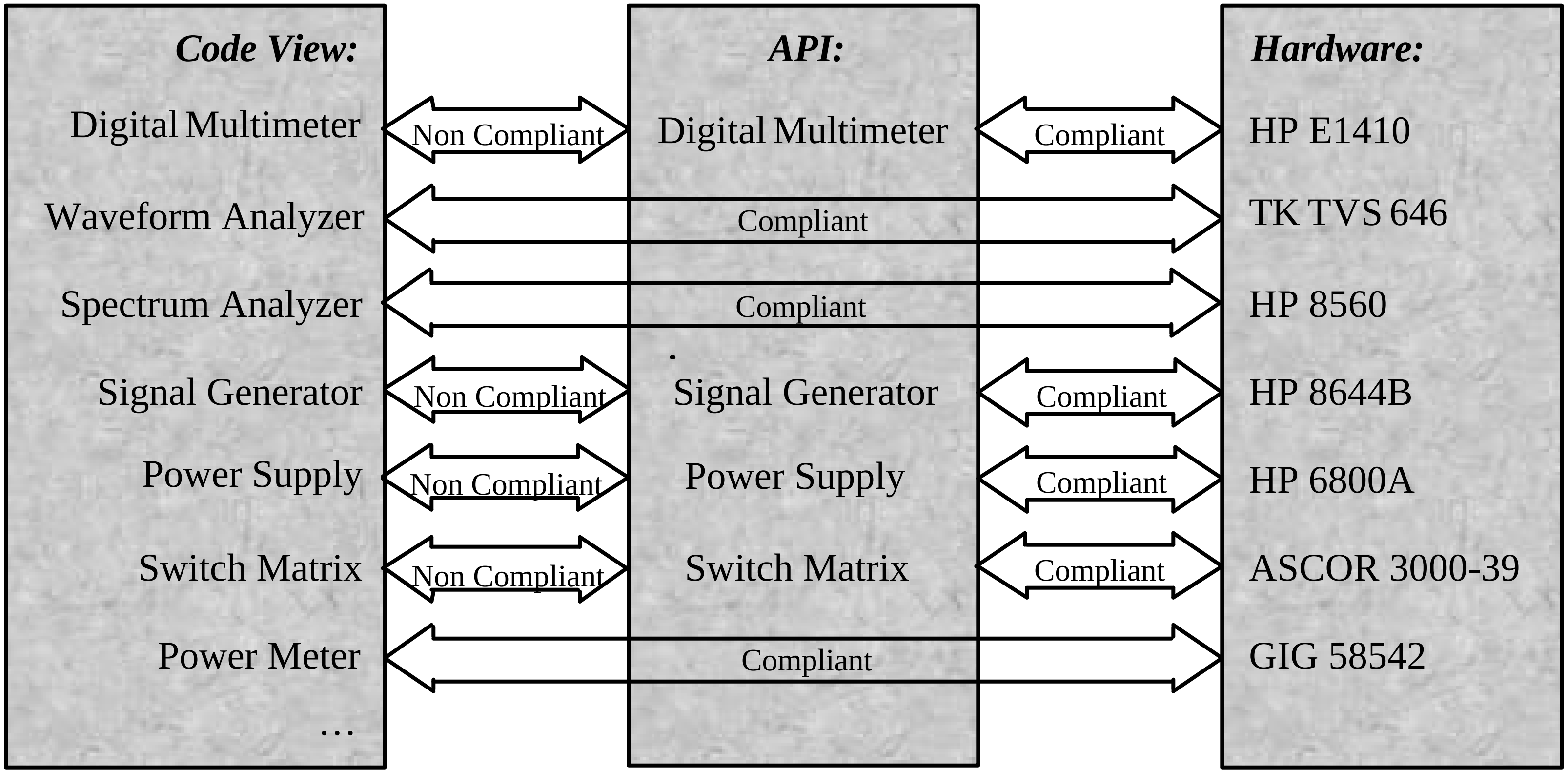
Test Structure (cont.)

Raytheon

Smart drivers apply here...

- **Smart Drivers**
  - **Remember their state**
  - **Cut bus communication to instruments**
  - **Speed up tests**

Application Program Interface



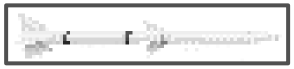
Code View:		*API:*		*Hardware:*
Digital Multimeter	Non Compliant	Digital Multimeter	Compliant	HP E1410
Waveform Analyzer		Compliant		TK TVS 646
Spectrum Analyzer		Compliant		HP 8560
Signal Generator	Non Compliant	Signal Generator	Compliant	HP 8644B
Power Supply	Non Compliant	Power Supply	Compliant	HP 6800A
Switch Matrix	Non Compliant	Switch Matrix	Compliant	ASCOR 3000-39
Power Meter		Compliant		GIG 58542
…				

Test Library

Raytheon

Hardware Configuration

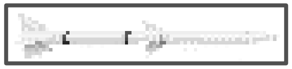
A



1 3 2 4 6 5 7 8 9 13 10 12 11 14 16 17 15 18

18 New Tests

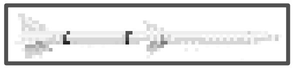
B



19 20 21 22 23 24 25

7 New Tests

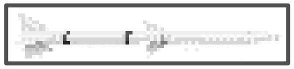
C



26 27 29 28

4 New Tests

D



30 31

2 New Tests

- Libraries of drivers
- Libraries of algorithms
- *Libraries of complete tests*

Summary

- A Common ATS Software Architecture Promotes
  - Reuse
  - Standardization
- Cooperation Development:
  - Reduces Cost
  - Shortens New and Upgrade ATS Development Schedules
  - Provides a Forum for Technology Sharing
- A Common ATS Software Architecture
  - Accelerates Time-to-market
  - Facilitates Standardization at the System Level
  - Enhances Interoperability Between Sites and Customers